USN | | | | | | | | | | |

INTERNAL ASSESSMENT TEST – I

Sub:	Microcontroller							Code:	BEC405A
Date:	24/03/25	Duration:	90 mins	Max Marks:	50	Sem:	IV	Branch:	ECE

Answer any 5 full questions

		Marks	CO	RBT
1	Explain the Internal RAM Memory Structure and Programming model of 8051	[10]	CO1	L2
2	Explain with functional block diagram 'Port 0' and 'Port 1'of 8051.	[10]	CO1	L2
3	(a) Compare microprocessor with microcontroller. (b) Name the bit addressable SFRs present in 8051 with addresses.	[04] [06]	CO1	L2
4	List and explain the special features of 8051 microcontroller.	[10]	CO1	L2

		Marks	CO	RBT
5	With a neat diagram, explain the steps to interface 32 KB of program RAM and 16 KB of ROM to 8051 based systems.	[10]	CO1	L3
6	Explain the arithmetic instructions of the 8051 microcontroller with suitable illustration.	[10]	CO2	L2
7	Identify the addressing modes for following instructions. a)ADD A,@R0 b) MOV R0, #45h c) MOV 40h, A d) MOV R3, A e) MOVC A, @A+PC Explain the operation of following . a) MOV DPTR, #2000H b) SUBB A, R1 c) SETB C d) ANL A, #0F0H e) CJNE A, #50H, LABEL	[10]	CO2	L2
8	Write an ALP to exchange contents of external data memory 8100h with contents of internal data memory 40h	[10]	CO2	L3

Solution

1. Explain the Internal RAM Memory Structure and Programming model of 8051

Working Registers

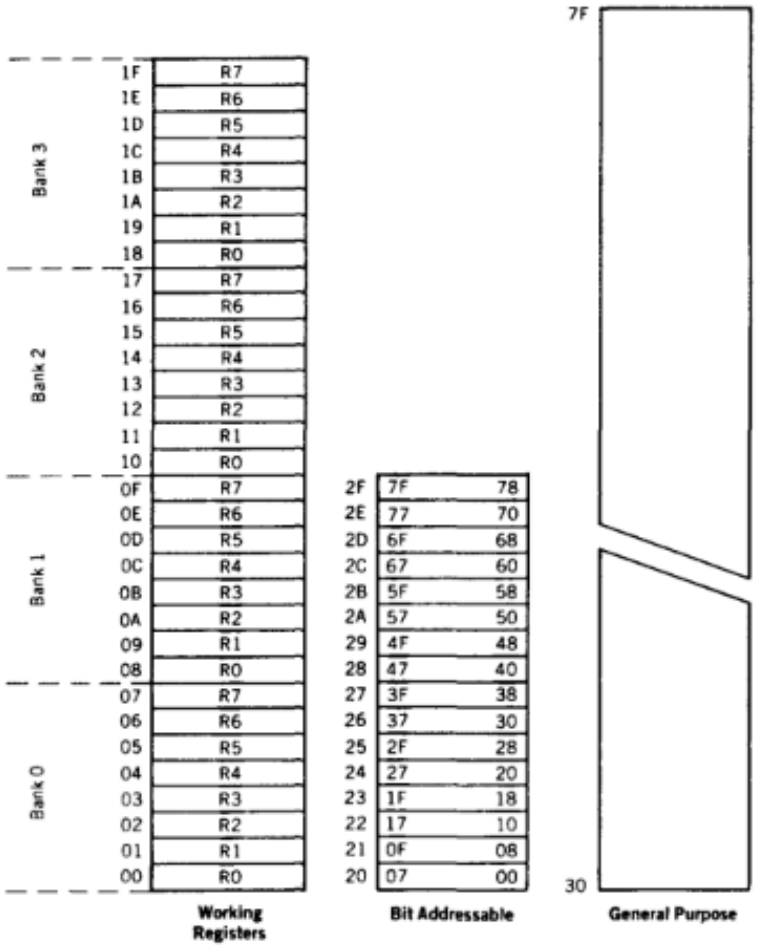
Register Banks: 00h to 1Fh. The 8051 uses 8 general-purpose registers R0 through R7 (R0, R1, R2, R3, R4, R5, R6, and R7). There are four such register banks. Selection of register bank can be done through RS1,RS0 bits of PSW. On reset, the default Register Bank 0 will be selected.

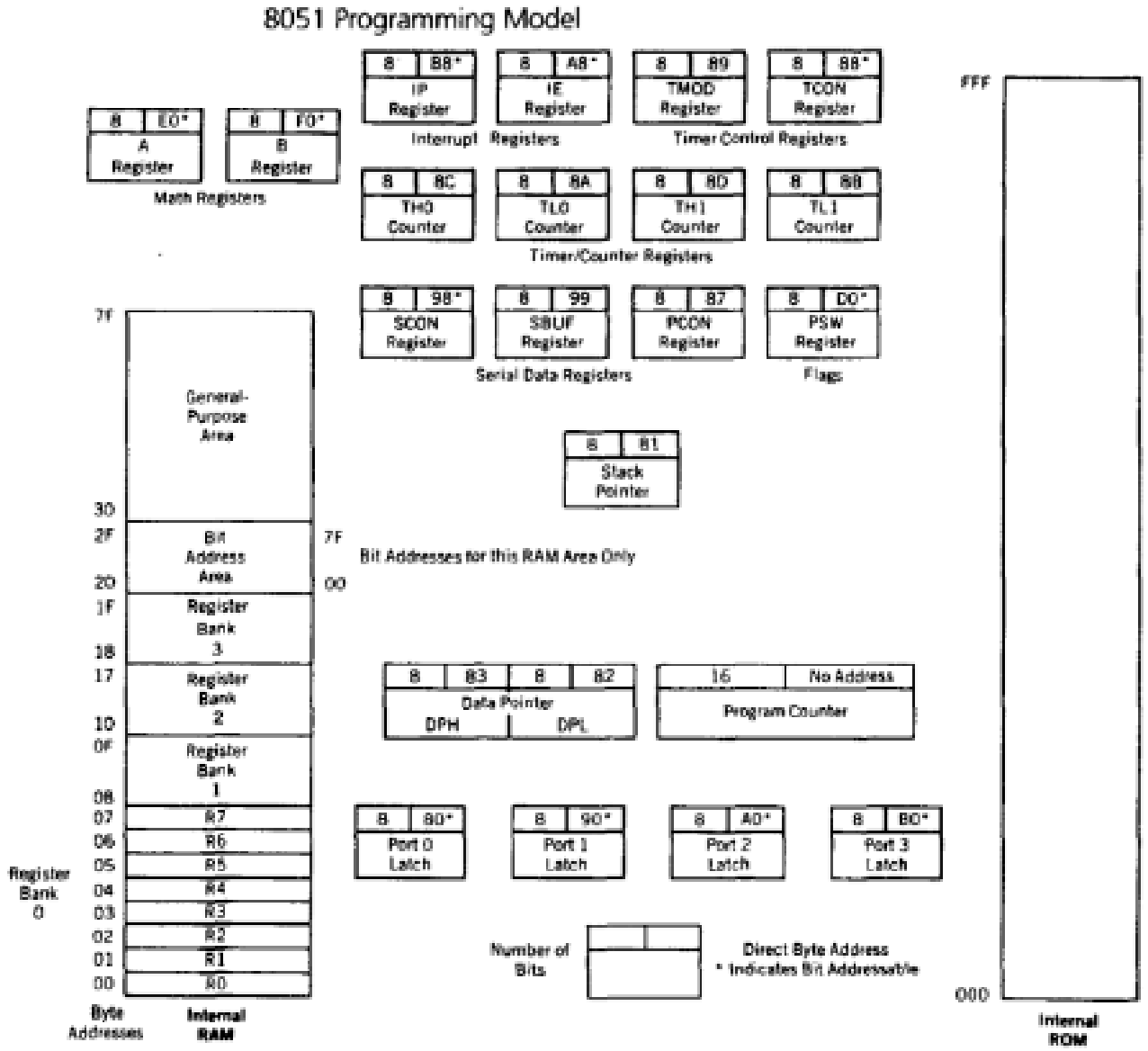
Bit Addressable RAM: 20h to 2Fh . The 8051 supports a special feature which allows access to bit variables. This is where individual memory bits in Internal RAM can be set or cleared. In all there are 128 bits numbered 00h to 7Fh. Being bit variables any one variable can have a value 0 or 1. A bit variable can be set with a command such as SETB and cleared with a command such as CLR. Example instructions are:

SETB 25h ; sets the bit 25h (becomes 1)
CLR 25h ; clears bit 25h (becomes 0)
Note, bit 25h is actually bit 5 of Internal RAM location 24h.
The Bit Addressable area of the RAM is just 16 bytes of Internal RAM located between 20h and 2Fh.

General Purpose RAM: 30h to 7Fh. Even if 80 bytes of Internal RAM memory are available for general-purpose data storage, user should take care while using the memory location from 00 -2Fh since these locations are also the default register space, stack space, and bit addressable space. It is a good practice to use general purpose memory from 30 – 7Fh. The general purpose RAM can be accessed using direct or indirect addressing modes.

Internal RAM Organization




8051 Programming Model
8 E0* A Register
8 F0* B Register
Math Registers
8 B8* IP Register
8 A8* IE Register
Interrupt Registers
8 89 TMOD Register
8 88* TCON Register
Timer Control Registers
8 8C TH0 Counter
8 8A TL0 Counter
8 8D TH1 Counter
8 8B TL1 Counter
Timer/Counter Registers
8 98* SCON Register
8 99 SBUF Register
Serial Data Registers
8 87 PCON Register
8 D0* PSW Register
Flags
8 81 Stack Pointer
8 83 8 82 Data Pointer DPH DPL
16 No Address Program Counter
8 80* Port 0 Latch
8 90* Port 1 Latch
8 A0* Port 2 Latch
8 B0* Port 3 Latch
Number of Bits
Direct Byte Address
* Indicates Bit Addressable
7F
30
2F
20
1F
18
17
10
0F
08
07
06
05
04
03
02
01
00
General-Purpose Area
Bit Address Area
Register Bank 3
Register Bank 2
Register Bank 1
R7
R6
R5
R4
R3
R2
R1
R0
Register Bank 0
7F
00
Bit Addresses for this RAM Area Only
Byte Addresses
Internal RAM
FFF
000
Internal ROM

The programming model of the 8051 in Figure 2.1b shows the 8051 as a collection of 8- and 16-bit registers and 8-bit memory locations. These registers and memory locations can be made to operate using the software instructions that are incorporated as part of the design. The program instructions have to do with the control of the registers and digital data paths that are physically contained inside the 8051, as well as memory locations that are physically located outside the 8051.

The model is complicated by the number of special-purpose registers that must be present to make a microcomputer a microcontroller. A cursory inspection of the model is recommended for the first-time viewer; return to the model as needed while progressing through the remainder of the text.

Most of the registers have a specific function; those that do occupy an individual block with a symbolic name, such as A or TH0 or PC. Others, which are generally indistinguishable from each other, are grouped in a larger block, such as internal ROM or RAM memory.

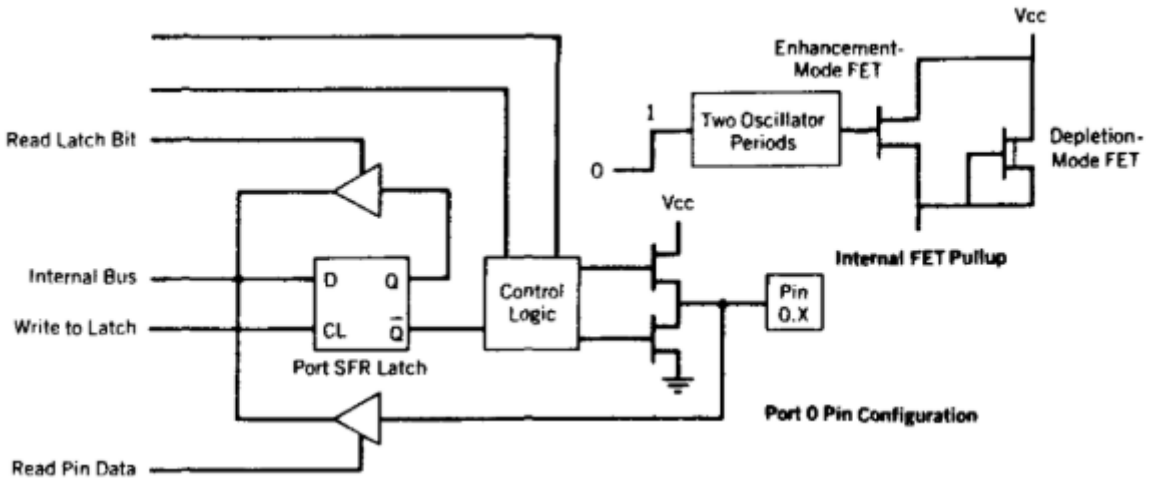
Each register, with the exception of the program counter, has an internal 1-byte address assigned to it. Some registers (marked with an asterisk * in Figure 2.1b) are both byte and bit addressable. That is, the entire byte of data at such register addresses may be read or altered, or individual bits may be read or altered. Software instructions are generally able to specify a register by its address, its symbolic name, or both.

A pinout of the 8051 packaged in a 40-pin DIP is shown in Figure 2.2 with the full and abbreviated names of the signals for each pin. It is important to note that many of the pins are used for more than one function (the alternate functions are shown in parentheses in Figure 2.2). Not all of the possible 8051 features may be used *at the same time*.

Programming instructions or physical pin connections determine the use of any multifunction pins. For example, port 3 bit 0 (abbreviated P3.0) may be used as a general-purpose I/O pin, or as an input (RXD) to SBUF, the serial data receiver register. The system designer decides which of these two functions is to be used and designs the hardware and software affecting that pin accordingly.

2. Explain with functional block diagram 'Port 0' and 'Port 1'of 8051.

PORT 0 configuration :



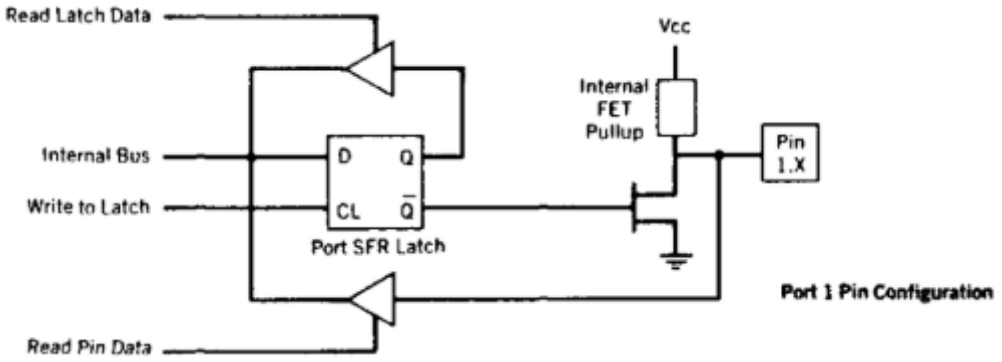
Port 0

Port 0 pins may serve as inputs, outputs, or, when used together, as a bi-directional low-order address and data bus for external memory. For example, when a pin is to be used as an input, a 1 *must be* written to the corresponding port 0 latch by the program, thus turning both of the output transistors off, which in turn causes the pin to "float" in a high-impedance state, and the pin is essentially connected to the input buffer.

When used as an output, the pin latches that are programmed to a 0 will turn on the lower FET, grounding the pin. All latches that are programmed to a 1 still float; thus, external pullup resistors will be needed to supply a logic high when using port 0 as an output.

When port 0 is used as an address bus to external memory, internal control signals switch the address lines to the gates of the Field Effect Transistories (FETs). A logic 1 on an address bit will turn the upper FET on and the lower FET off to provide a logic high at the pin. When the address bit is a zero, the lower FET is on and the upper FET off to provide a logic low at the pin. After the address has been formed and latched into external circuits by the Address Latch Enable (ALE) pulse, the bus is turned around to become a data bus. Port 0 now reads data from the external memory and must be configured as an input, so a logic 1 is automatically written by internal control logic to all port 0 latches.

PORT 1 configuration :



Port 1 Pin Configuration

Port 1

Port 1 pins have no dual functions. Therefore, the output latch is connected directly to the gate of the lower FET, which has an FET circuit labeled "Internal FET Pullup" as an active pullup load.

Used as an input, a 1 is written to the latch, turning the lower FET off; the pin and the input to the pin buffer are pulled high by the FET load. An external circuit can overcome the high impedance pullup and drive the pin low to input a 0 or leave the input high for a 1.

If used as an output, the latches containing a 1 can drive the input of an external circuit high through the pullup. If a 0 is written to the latch, the lower FET is on, the pullup is off, and the pin can drive the input of the external circuit low.

To aid in speeding up switching times when the pin is used as an output, the internal FET pullup has another FET in parallel with it. The second FET is turned on for two oscillator time periods during a low-to-high transition on the pin, as shown in Figure 2.7. This arrangement provides a low impedance path to the positive voltage supply to help reduce rise times in charging any parasitic capacitances in the external circuitry.

3.Compare microprocessor with microcontroller.

(c) Name the bit addressable SFRs present in 8051 with addresses

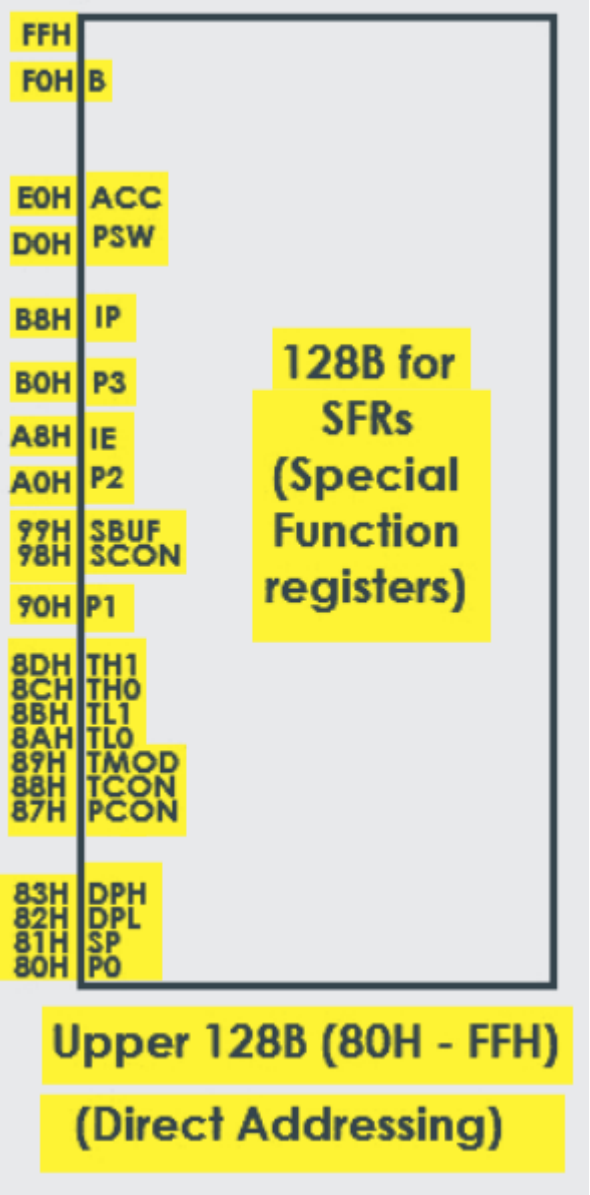
a)

Microprocessor contains ALU, General purpose registers, stack pointer, program counter, clock timing circuit, interrupt circuit	Microcontroller contains the circuitry of microprocessor, and in addition it has built in ROM, RAM, I/O Devices, Timers/Counters etc.
It has many instructions to move data between memory and CPU	It has few instructions to move data between memory and CPU
Few bit handling instruction	It has many bit handling instructions
Less number of pins are multifunctional	More number of pins are multifunctional
Single memory map for data and code (program)	Separate memory map for data and code (program)
Access time for memory and IO are more	Less access time for built in memory and IO.
Microprocessor based system requires additional hardware	It requires less additional hardwares
More flexible in the design point of view	Less flexible since the additional circuits which is residing inside the microcontroller is fixed for a particular microcontroller
Large number of instructions with flexible addressing modes	Limited number of instructions with few addressing modes

b)Name the bit addressable SFRs present in 8051 with addresses

Special Function Registers in 8051 Microcontroller are used to program and control different hardware peripherals like Timers, Serial Port, I/O Ports etc. In fact, by manipulating the SFR in 8051 Microcontroller, you can assess or change the operating mode of the 8051 Microcontroller.

As a reminder, the following image shows you the basic structure of 8051 Microcontroller's Internal RAM.



Out of these 128 Memory Locations (80H to FFH), there are only 21 locations that are actually assigned to SFRs. Each SFR has one Byte Address and also a unique name which specifies its purpose.

Since the SFRs are a part of the Internal RAM Structure, you can access SFRs as if you access the Internal RAM. The main difference is the address space: first 128 Bytes (00H to 7FH) is for regular Internal RAM and next 128 Bytes (80H to FFH) is for SFRs.

Name of the Register	*Function*	*Internal RAM Address (HEX)*
ACC	Accumulator	E0H
B	B Register (for Arithmetic)	F0H
DPH	Addressing External Memory	83H
DPL	Addressing External Memory	82H
IE	Interrupt Enable Control	A8H
IP	Interrupt Priority	B8H
P0	PORT 0 Latch	80H
P1	PORT 1 Latch	90H
P2	PORT 2 Latch	A0H
P3	PORT 3 Latch	B0H
PCON	Power Control	87H
PSW	Program Status Word	D0H
SCON	Serial Port Control	98H
SBUF	Serial Port Data Buffer	99H
SP	Stack Pointer	81H
TMOD	Timer / Counter Mode Control	89H
TCON	Timer / Counter Control	88H
TL0	Timer 0 LOW Byte	8AH
TH0	Timer 0 HIGH Byte	8CH
TL1	Timer 1 LOW Byte	8BH
TH1	Timer 1 HIGH Byte	8DH

4.List and explain the special features of 8051 microcontroller.

Salient features of 8051 microcontroller are given below.

- Eight bit CPU
- On chip clock oscillator
- 4Kbytes of internal program memory (code memory) *[ROM]*
- 128 bytes of internal data memory *[RAM]*
- 64 Kbytes of external program memory address space.
- 64 Kbytes of external data memory address space.
- 32 bi directional I/O lines (can be used as four 8 bit ports or 32 individually addressable I/O lines)
- Two 16 Bit Timer/Counter :T0, T1
- Full Duplex serial data receiver/transmitter
- Four Register banks with 8 registers in each bank.
- Sixteen bit Program counter (PC) and a data pointer (DPTR)
- 8 Bit Program Status Word (PSW)
- 8 Bit Stack Pointer

- Five vector interrupt structure (RESET not considered as an interrupt.)
- 8051 CPU consists of 8 bit ALU with associated registers like accumulator 'A' , B register, PSW, SP, 16 bit program counter, stack pointer.
- ALU can perform arithmetic and logic functions on 8 bit variables.
- 8051 has 128 bytes of internal RAM which is divided into
    - Working registers [00 – 1F]
    - Bit addressable memory area [20 – 2F]
    - General purpose memory area (Scratch pad memory) [30-7F]

5. With a neat diagram, explain the steps to interface 32 KB of program RAM and 16 KB of ROM to 8051 based systems.

Address Decoding (Memory Map) for 16k x 8 ROM.

Address	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	HEX adrs.
starting	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
end	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3FFFH

Address Decoding (Memory Map) for 32k x 8 RAM.

Address	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	HEX adrs.
starting	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000H
end	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	FFFFH

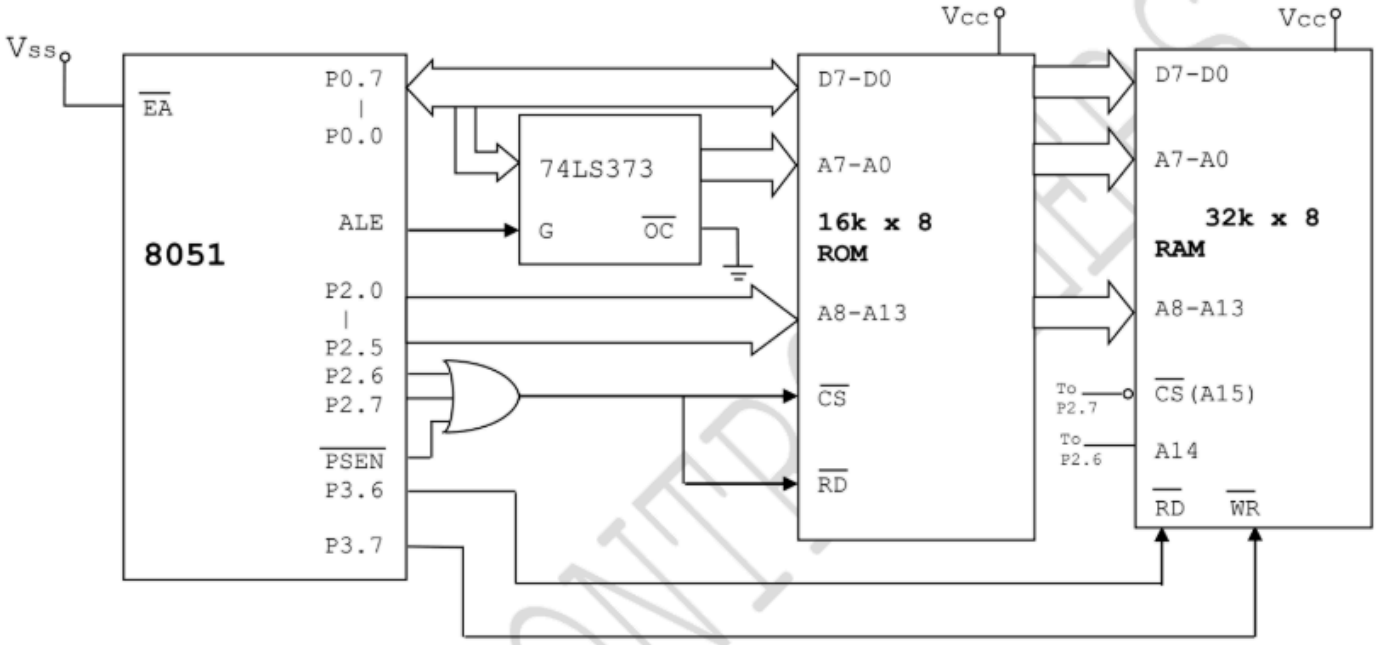


FIGURE 7 16K X 8 ROM AND 32K X 8 RAM INTERFACING TO µC 8051.

6.Explain the arithmetic instructions of the 8051 microcontroller with suitable illustration.

Mnemonics
ADD A, Rn

ADD A, a8
ADD A, @Ri
ADD A, #d8
ADDC A, Rn
ADDC A, a8
ADDC A, @Ri
ADDC A, #d8
SUBB A, Rn
SUBB A, a8
SUBB A, @Ri
SUBB A, #d8
INC A
INC Rn
INC a8
INC @Ri
DEC A
DEC Rn
DEC a8
DEC @Ri
INC DPTR
MUL AB
DIV AB
DA A

Examples

Sr.No	Instruction & Description

1	**ADD A, 32H** This is an instruction of type ADD A, #d8. The immediate data 32H is added to register A. The result is also stored in A.
2	**ADDC A, @R1** This is an instruction of type ADDC A, @Ri. It means the content on internal RAM location which is pointed by register R1 is added to A.
3	**SUBB A, R5** This is SUBB A, Rn type instruction. The SUBBstands for Subtract with borrow. So the content of R5 will be subtracted from A.
4	**INC 40H** This is a type of INC a8. Here the content in local RAM whose address is 40H, it is increased by 1.
5	**INC DPTR** It can increase the content of DPTR by 1. This instruction takes two machine cycle to execute.
6	**MUL AB** This instruction is used to multiply the content of register A and B. The 16-bit address will be stored at B and A registers. The B will hold the MS byte, and A will hold the LSByte.
7	**DIV AB** This instruction is used to divide the content of A register by B register. The 8-bit quotient is stored into the register A, and the 8-bit remainder is stored into the register B.

7. Identify the addressing modes for following instructions.
a)ADD A,@R0 Indirect register addressing mode

b) MOV R0, #45h - Immediate addressing mode

c) MOV 40h, A - Direct addressing mode

d) MOV R3, A - Register addressing mode

e) MOVC A, @A+PC - Indexed addressing mode

Explain the operation of following .

a) MOV DPTR, #2000H - Move value 2000h to Data pointer

b) SUBB A, R1 - Subtract R1 contents from Accumulator with borrow

c) SETB C - Sert bit carry flag

d) ANL A, #0F0H - AND operation of accumulator contents with 0F0 h

e) CJNE A, #50H, LABEL - Compare Acc contents with data 50h , if not equal go to label.

8.Write an ALP to exchange contents of external data memory 8100h with contents of internal data memory 40h

```
ORG 0000H
MOV R0, #40H            ; load R0 with Source address 40h in internal memory
MOV DPTR, #8100H        ; load dptr with destination address 8100h external memory
```

```
MOV R1, #06H            ; load the no of bytes counter value in R1
LOOP:MOVX A, @DPTR      ;Move the content pointed by DPTR into Accumulator
MOV B, @R0              ;Move the content pointed by R0 into B register
XCH A, B                ; exchange the contents between A and B
MOVX @DPTR, A           ; load the accumulator content into address pointed by DPTR
MOV @R0, B              ; load the content of B register into address pointed by R0
INC DPTR                ; increment DPTR
INC R0                  ;increment R0
DJNZ R1, LOOP           ;check if the counter value is zero, else go to LOOP
SJMP $
END
```